(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 0 774 853 A2

(12) EUROPEAN PATENT APPLICATION

(43) Date of publication:
21.05.1997 Bulletin 1997/21

(51) Int. Cl.[6]: **H04M 3/50**, G06F 9/00

(21) Application number: **96116616.2**

(22) Date of filing: **16.10.1996**

(84) Designated Contracting States:
DE FR GB

(30) Priority: **15.11.1995 US 558322**

(71) Applicant: **MICROSOFT CORPORATION**
Redmond, Washington 98052-6399 (US)

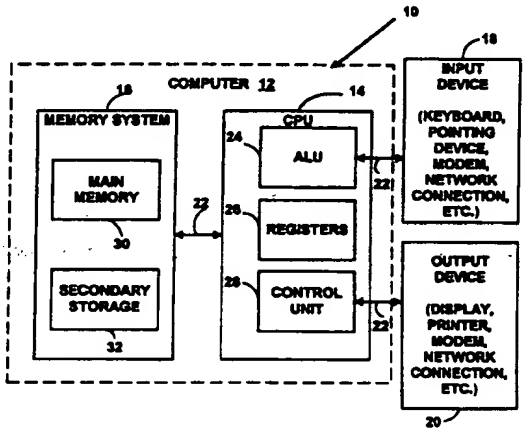
(72) Inventor: **Fawcet, Phillip E.**
Washington 98019 (US)

(74) Representative: **Meddle, Alan Leonard**
FORRESTER & BOEHMERT
Franz-Joseph-Strasse 38
80801 München (DE)

(54) Method and system for graphically displaying and navigating through an interactive voice response menu

(57) An interactive voice response menu for computer software support which typically presents voice queries to a user is graphically displayed on a user computer. The interactive voice response menu is graphically displayed using a hierarchical paradigm scheme which is created based on statistics gathered from known user problem and is updated continuously when additional user problems are discovered. The user navigates through the graphical interactive voice response menu hierarchical scheme choosing more and more specific queries with the user computer to help diagnose the user's software problem. Also, when the user visits selected levels in the hierarchical scheme, diagnostic applications are launched automatically on the user computer which gather data for use in diagnosing the user's computer software problem.

FIG. 1



EP 0 774 853 A2

Description

FIELD OF INVENTION

The present invention relates to the support of computer software products. More specifically, it relates to a method of graphically displaying and navigating through an interactive voice response menu of queries used to diagnose problems encountered using computer software.

BACKGROUND AND SUMMARY OF THE INVENTION

Computer software can pose a number of difficult and frustrating problems for a user. Computer software often contains defects, which can prevent a user from fully utilizing the computer software. A user may also have trouble installing the computer software, have trouble using the computer software, etc. When problems arise with computer software, a user will often call a support center. When a support center is called, the user is typically first presented with more information from an interactive voice response unit (IVRU). The interactive voice response unit poses voice queries to the user. The user inputs responses (e.g. with a touch tone telephone) to the voice queries, and then is presented additional voice queries based on the responses. The user responses to the voice queries are typically used to direct a user to a support engineer with the proper expertise to help solve a user's problem.

The interactive voice response unit presents a user with voice queries based on some hierarchical scheme (e.g. a decision tree). General voice queries are presented first, and then, based on user responses, more specific queries are presented to help narrow and/or focus the user's problem. For example, if the interactive voice response unit uses a decision tree, a user would be presented first with queries from the "root"(i.e. the top or most general level) level of the decision tree. Then based on user responses, the user is presented with more and more specific voice queries (i.e. intermediate level queries). Finally, the user would reach the lowest level in the decision tree. At the lowest level in the decision tree, the "leaf" level, the user may be presented with voice information that may help solve the user's problem. The user may also be asked to hold for the next available support engineer.

As an example, if the interactive voice response unit was set up as a decision tree, the interactive voice response at the root level might be, "Press '1' for network problems, Press '2' for operating system problems, Press '3' for printer problems, Press '4' for spreadsheet problems," etc. If a user was having printer problems, after pressing '3' at the root level, the interactive voice response unit responds with a second level (intermediate level) of query choices such as "Press '1' if your printer won't print at all, Press '2' if your printer is printing unrecognizable characters, Press '3' if your printer is not printing all the information on one page," etc. This process would be repeated until the user reached a leaf level of the decision tree. At the leaf level, the user may be given additional information that may solve the problem (e.g. your printer driver seems to be out-of-date), or is told to stay on the line to receive help from a support engineer. Which leaf node a user reaches determines which support engineer the user will speak with. The user responses to the voice queries are also used to make a preliminary diagnosis of the user problem.

There are several problems associated with using an interactive voice unit based on a hierarchical scheme, such as decision tree, to help diagnose a user's software problem. A user may have to listen to all of the voice queries at the top level of hierarchal scheme before making even a first selection. This is a time consuming process, and the user may have to repeat the general level voice query messages several times before a choice is made. At levels below the top level (intermediate levels), the user is faced with a number of decisions and may not know which choice is appropriate based on the brief voice menu description. If the user makes several choices based on the voice queries and ends up at the wrong place in the hierarchical scheme, then the user must "unwind" the choices typically by starting over at some higher level (e.g. the top level), and then descending again in the hierarchical scheme by making new choices. This wastes a considerable amount of the user's time and leads to user dissatisfaction.

Interactive voice response units are also typically set up to contain very general information to serve a large number of potential users with a wide variety of computer software problems. Many times the voice queries are not specific enough to allow a user to adequately diagnose a problem without speaking to a support engineer. When a user spends too much time navigating in the voice response menu, the user's problem will not be narrowed or categorized by the time a support engineer is available. The user may also forget, or may not understand the brief general messages that are recorded for each level in the voice response menu and may make the wrong decision several times and have to start over. As a result, the user may become frustrated and hang up.

In accordance with the illustrated embodiment of the present invention some of the problems associated with using an interactive voice response unit to diagnose user problems are solved. As one of the first voice queries presented in the interactive voice response menu, a user is presented with a voice option to create a graphical display of the interactive voice response queries on the user computer. If this option is chosen by the user, the user is presented with a graphical display of the information contained in the interactive voice response menu. The user responses are then made from the user computer.

In the illustrated embodiment, the graphical display presents the interactive voice response information using a hierarchal paradigm based scheme. Unlike prior

interactive voice response decision trees, the hierarchal scheme is based on data collected from problems previously encountered by users and is continually updated as users encounter new or previously unknown problems with computer software. Further, the queries presented to a user are determined using a probabilistic scheme which directs the user to solutions for problems that are most likely to be encountered using particular computer software.

The graphical display allows the user to navigate up and down in the hierarchal paradigm scheme quickly and easily. The user can also "zoom in" and "zoom out" on a part of the hierarchal scheme, skipping intermediate steps that would require making responses to voice queries. If the user becomes confused as to which query to choose, there is a decision advisor available with information to help the user make the proper choice. The user also has the option at any time of switching to a voice mode to listen to the voice queries.

In another embodiment of the invention, when the user reaches designated levels in the hierarchal scheme, the support center computer will launch diagnostic applications to further diagnose the user's problem. This can be done before the user is connected to a support engineer. The data gathered from the user computer is automatically transferred to the support center and displayed for the support engineer to whom the user's call is routed. As a result, by the time a user is connected with a support engineer, the user's problem has been thoroughly diagnosed and the support engineer already has diagnostic data to help solve the user's problem.

The graphical display of the interactive voice response decision information may save the user a substantial amount of time. Since the voice menu choices are graphically displayed, the user can step immediately to any level in the hierarchical scheme without having to listen to a number of voice queries. If the user makes a wrong choice, the user can easily undo the incorrect choice by choosing a new level in the hierarchal scheme, without having to visit intermediate levels. Thus, by using a graphical display of the interactive voice menu, the user's problem may be diagnosed more quickly and thoroughly than it could be by using other methods such as an interactive voice response menu or voice interaction with a support engineer.

The graphical display of the interactive voice response information also saves the software creator a significant amount of support costs. The graphical queries presented to the user are kept up-to-date based on known problems with the computer software. The user may find a solution to a particular problem by using graphical display and not need, or want to talk with a support engineer. If the user's problem is not solved by using the graphical display, a display of the data gathered by the system is presented to the support engineer at the outset of a conversation with a user. As a result, the support engineer typically has to spend far less time on the phone with a user, in order to diagnose the user's problem.

The foregoing and other features and advantages of the illustrated embodiment of the present invention will be more readily apparent from the following detailed description, which proceeds with reference to the accompanying drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a block diagram of a computer system used to implement an illustrated embodiment of the present invention.

FIG. 2 is a block diagram showing a support center and user computers.

FIG. 3 is a diagram showing how a user and a support engineer would interact.

FIG. 4 is a block diagram showing a layered software support messaging architecture used to produce a graphical display.

FIG. 5 is a block diagram showing greater detail of the software messaging architecture used to produce the graphical display.

FIGS. 6A-6B are block diagrams showing the menu boxes and the choice boxes of a graphical display of an interactive voice response menu.

FIG. 7 is a block diagram showing a menu box and a choice box at one of the lowest levels in the hierarchical scheme of a graphical display of an interactive voice response menu.

FIG. 8 is a block diagram showing the decision advisor for the display information shown in FIG. 5B.

DETAILED DESCRIPTION OF AN ILLUSTRATED EMBODIMENT

Referring to FIG. 1, an operating environment for the preferred embodiment of the present invention is a computer system 10 with a computer 12 that comprises at least one high speed processing unit (CPU) 14, in conjunction with a memory system 16, an input device 18, and an output device 20. These elements are interconnected by a bus structure 22.

The illustrated CPU 14 is of familiar design and includes an ALU 24 for performing computations, a collection of registers 26 for temporary storage of data and instructions, and a control unit 28 for controlling operation of the system 10.

Any of a variety of processors, including those from Digital Equipment, Sun, MIPS, IBM, Motorola, NEC, Intel, Cyrix, AMD, Nexgen and others are equally preferred for CPU 14. Although shown with one CPU 14, computer system 10 may alternatively include multiple processing units.

The memory system 16 includes main memory 30 and secondary storage 32. Illustrated main memory 30 is high speed random access memory (RAM) and read only memory (ROM). Main memory 30 can include any additional or alternative high speed memory device or memory circuitry. Secondary storage 32 takes the form

of long term storage, such as ROM, optical or magnetic disks, organic memory or any other volatile or non-volatile mass storage system. Those skilled in the art will recognize that memory 16 can comprise a variety and/or combination of alternative components.

The input and output devices 18, 20 are also familiar. The input device 18 can comprise a keyboard, mouse, pointing device, sound device (e.g. a microphone, etc.), or any other device providing input to the computer system 10. The output device 20 can comprise a display, a printer, a sound device (e.g. a speaker, etc.), or other device providing output to the computer system 10. The input/output devices 18, 20 can also include network connections, modems, or other devices used for communications with other computer systems or devices.

As is familiar to those skilled in the art, the computer system 10 further includes an operating system and at least one application program. The operating system is a set of software which controls the computer system's operation and the allocation of resources. The application program is a set of software that performs a task desired by the user, making use of computer resources made available through the operating system. Both are resident in the illustrated memory system 16.

In accordance with the practices of persons skilled in the art of computer programming, the present invention is described below with reference to symbolic representations of operations that are performed by computer system 10, unless indicated otherwise. Such operations are sometimes referred to as being computer-executed. It will be appreciated that the operations which are symbolically represented include the manipulation by CPU 14 of electrical signals representing data bits and the maintenance of data bits at memory locations in memory system 16, as well as other processing of signals. The memory locations where data bits are maintained are physical locations that have particular electrical, magnetic, optical, or organic properties corresponding to the data bits.

As is shown in FIG. 2, the illustrated embodiment of the invention consists of a support center 34 with one or more support computer systems 36 (e.g. the computer system that was described in FIG. 1) and a plurality of user computer systems 38. If a plurality of support computers are used, then the support computers may be connected by a local area network (LAN) 40 or any other similar connection technology. However, it is also possible for a support center to have other configurations. For example, a smaller number of larger computers (i.e. a few mainframe, mini, etc. computers) with a number of internal programs or processes running on the larger computers capable of establishing communications links to the user computers. The support center provides a plurality of communications links 42, such as telecommunications connections, (e.g. modem connections, ISDN connections, ATM connections, frame relay connections, etc.), network connections, (e.g. Internet, etc.), satellite connections (e.g. Digital Satellite Services, etc.), wireless connections, two way paging connections, etc. to allow one or more user computers to simultaneously connect to the support computer(s). The communications links are each connected to one or more interactive voice response units (IVRU) 44 that can be accessed by user computers.

The IVRUs 44 are connected to an IVRU server 46. The IVRU server permits a user to bypass the IVRU menu and directly access one or more databases 48 maintained at the support center. The support center maintains one or more databases 48 that are used to store database entries consisting of problems previously encountered by other users for the computer software supported by the support center. When a user chooses to bypass the IVRU, the IVRU server permits to the user to query the support center databases to potentially find a solution to the problem the user is encountering that may not be mentioned in the IVRU menu.

In addition, a subset of the database entries, consisting of problems previously encountered by other users are used to create the audio menus in the IVRU. The same database entries are also used to create a graphical display of the IVRU audio menus.

The illustrated embodiment of the invention is implemented in the Windows 95™ operating system by Microsoft Corporation of Redmond, Washington using VoiceView™ data transfer technology by Radish Communications Systems of Boulder, Colorado. Under VoiceView, only voice OR data is transmitted at any one particular instance in time (i.e. VoiceView™ modems are alternating voice and data (AVD) modems). The invention can likewise be practiced with other operating systems and other technologies that allow simultaneous voice AND data transfer, such as simultaneous voice and data (SVD) modems, ISDN devices, etc. or with network or other connections. The VoiceView™ data protocol technology has been enhanced to provide functionality not available in the standard VoiceView™ data protocol as will be explained below.

As is shown in FIG. 3, when a user 50, using a user computer 38 calls a support center 34 (hereinafter referred to as a product support service (PSS) center) with a VoiceView™ modem 52 (or other appropriate communications device) and wants help with a particular computer software problem, a communications path 54 (via a socket connection 56) for data transfer is set up over the same voice line which carries the user's voice signals. The communications path is routed through an interactive voice response unit (IVRU) 44 in the PSS center. The IVRU is connected to an IVRU server 46. A VoiceView™ modem allows the user to make a voice connection with a support center, and allows the support center to send/receive data to/from a user's computer over the same telephone line as the voice connection.

The PSS system is implemented using VoiceView modems 52 as is shown in FIG. 4. However, other

devices capable of providing two-way voice/data communications could also be used. The VoiceView modems 52 establish a socket connection 56 that is used for communications between a client (PSS computer) 36 and a server (user computer) 38. As is shown in FIG. 4, the PSS system implements a two-layer protocol stack: the communication layer 58 and the application layer 60. The communication layer's 58 primary responsibility is to transfer data between the client 36 and the server 38. The client 36 represents a PSS computer and the server 38 represents the user's computer. The two layer protocol stack provides a common network protocol for passing data and issuing commands between the client 36 and server 38.

The application layer's 60 primary purpose is to utilize the communication layer's network services while maintaining ignorance of the networks underlying protocol and hardware. The PSS's application programs 62 reside in the application layer 60.

The two-layered protocol approach frees the application program 62 from the communication protocol complexity and implementation. The two-layer protocol stack operates over a socket connection. As is well known in the art, a socket is a communication object from which messages are sent and received. Sockets are common inter-process communication objects in many operating systems (e.g. 4.x BSD UNIX®, Windows 95™, etc.).

Whether the messaging system is based on VoiceView™ sockets, UNIX® sockets, or datagrams, the protocol details are entirely hidden from the application layer 60. Therefore, the PSS application programs 62 will not have to be rewritten if the underlying communication protocol is changed. Also, socket management code need not be replicated in each of the application programs 62. Another benefit of this architecture is that any number of client diagnostic programs (on the PSS computers) can communicate with their corresponding server diagnostic programs (on the user computers) through this common protocol. Client and server applications do not communicate directly with each other because replacement of the network platform and/or network protocol would make the diagnostic applications obsolete.

The PSS messaging system diagnostic applications have two components: diagnostic interpreters (DIs) 64 and diagnostic agents (DAs) 66 as is shown in FIG. 5. The diagnostic interpreter 64 is client software which initiates diagnostic commands and interprets their results for the PSS. The diagnostic agent 66 is a server software which executes commands issued by the diagnostic interpreter and then returns results to the PSS.

The PSS messaging system has both client and server components in the communications layer 58 that are used for message transfer. The support client (SC) 68 is a client messaging process which sends and receives messages on behalf of diagnostic interpreters 64. The support agent (SA) 70 is a server messaging process which sends and receives messages on behalf of diagnostic agents 66. Also included in the communications layer 58 of both the client and the server are registers 72 and 74, respectively, used to keep track of which client/server processes are sending/receiving messages. Communication between a particular interpreter and agent occurs via a socket identifier allocated to the pair. Socket setup (**create()**, **bind()**, **listen()**), connection (**connect()**, **accept()**), and communication (**send()** **receive()**) are well known and will be understood by those skilled in the art.

After a socket connection 56 is established, a user is presented with interactive voice response menu queries from the IVRU 44. The interactive voice response system is built using a hierarchal paradigm from the information contained on a database 48. The hierarchal paradigm is based on data collected from problems previously encountered by users and is continually updated as users encounter new or previously unknown problems with computer software. Further, the queries presented to a user are determined using a probabilistic scheme which directs the user to solutions for problems that are most likely to be encountered using a particular set of computer software.

As an example, if a defect has been discovered with a printer driver that will cause every user with a particular printer to experience the same problem, it is highly probable that a majority of the calls a support center will receive for printer problems will relate directly to the defective driver. As a result, the interactive voice response unit will present voice queries that will point a user directly to the defective driver problem. Thus, the intermediate steps that a user would normally go through are skipped, as statistics suggests that a majority of the printer problems will be related to the defective driver.

Other potential problems, such as network problems, application problems, etc. can be presented to the user in a similar fashion.

In the illustrated embodiment, the first voice query presented by the interactive voice response unit to a user will be a query that will allow a user to graphically display the interactive voice response menu queries. Since the user has established a two-way connection between the user computer and the PSS computer, (e.g. via a socket connection 56), the PSS computer will launch a Windows 95™ diagnostic interpreter 64 (FIG. 5) (e.g. called the Voice Menu Graphical Interpreter) to help create a graphical display of the interactive voice response information. The diagnostic interpreter will download, register, and execute a diagnostic agent, (e.g. called the Voice Menu Graphical Agent), which will create the graphical display on the user computer from the interactive voice response menu information. Those skilled in the art will understand the downloading, registering, and execution of a software application in a client/server environment.

When the Voice Menu Graphical Agent is executed, it creates a graphical display of the interactive voice

response menu hierarchical paradigm, an example of which is shown in FIG 6A. However, the interactive voice response menu queries can be displayed with text, graphics, and/or a combination of text and graphics depending on user preference.

As an example, suppose a user is using a word processing program and has a problem printing a document. FIG. 6A shows a representation of the graphical display that could be presented to a user's computer. The interactive voice response menu queries are presented on the left side of the screen in a menu box 76, and choices (e.g. with a mouse, keyboard, etc. connected to the user computer) made by a user are presented on the right side of the screen in a choice box 78. Since the user is having problems using a word processing program, option 3 from the menu 76c is chosen from the menu box 76. As a user makes choices, the choices are displayed in the choice box 78. Since the user is at the top or most general level of the text menu in the hierarchical paradigm scheme, and no choices have yet been made, the choice box 78 is currently empty.

FIG. 6B shows the display after a user has chosen option 3 ("problems with a word processing program") 76c from FIG. 6A. A new level of queries is presented in the menu box 80 (80a-80d) on the left side of the screen, and the choice box 82 on right side of the screen displays the user choice 82a. The choice box 82 has only one entry 82a since the user has only made one choice. The number one (1:) in the upper left hand corner of box 82a indicates that this entry was chosen from level one in the interactive voice response menu hierarchical scheme. Query number three ("3. Problems with a word processing program") was the actual choice made at level one, as is shown in box 82a.

This scenario continues, user choices, new queries in the menu box, and new entries in the choice box until a user reaches the lowest possible level based on their choices in the hierarchical scheme. As a user moves "down" in the hierarchical scheme, entries are added to the choice box. As a user moves "up" in the hierarchical scheme, entries are removed from the choice box. As a result, the user always knows what choices have been made, and can immediately "zoom in" or "zoom out" to any level in the hierarchical scheme.

FIG. 7 shows the display after a user has reached one of the lowest levels in the hierarchical scheme. The menu box 84 does not contain any further queries, but now contains information which provides a potential solution to the user's problem (In the actual implementation, the information at the lowest level in the hierarchical scheme may be much more extensive than is shown in the example in box 84). The choice box 86 is filled with choices (86a-86d) the user has made at each level (e.g. box 86c shows on level three ("3:"), the user chose the second option "2. Problems printing with a laser printer").

If the user wishes to change levels in the hierarchical scheme at any time, the user simply chooses the appropriate entry in the choice box. For example, if the user has the display shown in FIG. 7, and wished to reach the general menu (highest level) in the hierarchical scheme, the user would simply choose (e.g. with a mouse or keyboard, etc.) the entry labeled one (i.e. 1:) containing the text "3. problems with a word processing program" 86a in the choice box. The user would then jump back to the display shown in FIG. 6A, skipping all levels in between. The menu box 74 (FIG. 6A) would be re-displayed, and the choice box 76 (FIG. 6A) would empty again since the user has returned to the highest level in the hierarchical scheme.

Jumping to intermediate levels in the hierarchical scheme would produce similar results, with the menu box displaying possible choices and the choice box partially filled with choices the user has made to get to the intermediate levels. The user can skip any or all of the intermediate steps that would be required when using the voice mode of an IVRU. For example, if the user made seven choices, the user would be down at least seven levels in the hierarchical paradigm. If the user then wished to return to level three, the user could jump immediately back to level three, by selecting "3:" from the choice box and then make new choices from level three. Using the IVRU in voice mode, the user would most likely be required to return to level one, make the same choices as made previously to return to level three, and then make new or additional choices, which would waste a considerable amount of time.

In addition, the user can jump out of the IVRU graphical menu at any time and use the IVRU server to access the support center data base. This allows the user to obtain additional information not displayed on the IVRU graphical menu that might help solve a particular problem.

FIGS. 6A, 6B, and 7 show an example of a textual display of the interactive voice response menu. However, any other display layout that would convey the information from the interactive voice menu hierarchical scheme could also be used.

A graphical/text combination on a graphical display can also be used. For example, the hierarchical display could be represented by a graphical "tree" structure containing many nodes. The user would then choose a "node" in the tree (analogous to choosing an item from the menu box). The tree node would then be expanded to present the "subtree" below the chosen node (analogous to a new menu box). After each user choice, a smaller scale display tree (analogous to the choice box) graphically shows the user path (e.g. in a color different from the rest of the tree) and present location in the tree. The node choosing is repeated until a "leaf" node is reached, where information is presented to potentially solve the user problem as described above (and shown in FIG. 7). Any number of other graphics/text schemes could also be used to display the interactive voice menu hierarchical scheme for the user.

If a user should be confused about which path in the hierarchical scheme to choose, a decision advisor

associated with each level gives a more detailed explanation to aid the user in making a correct decision. The decision advisor is more than a generic help function. The decision advisor knows what choices the user has made. Based on these choices, the decision advisor queries the support center databases and directs the user to make additional choices that, based previously encountered problems, are most like to solve the current user's problem. A decision advisor text is shown in FIG. 8. Suppose a user was having trouble printing a document from a word processing program. If the user had trouble deciding whether to choose option 2 (88b), or option 3 (88c), (i.e. trouble deciding whether their problem falls within "problems using a word processing program" or "problems printing with a word processing program") the decision advisor is used to help the user make the decision. The decision advisor "knew" the user was having problems with a word processing program (90a). After a query to the PSS database, the decision advisor would direct the user to the proper choice 92.

If the user invokes the decision advisor for option 3 (88c), an advisor box 92 is displayed with additional information to aid the user in making a decision based on the user's problem. In an actual implementation, the help information displayed by the decision advisor may be more extensive than is shown in 92.

In the worst case scenario, the user would reach the lowest level in the hierarchical paradigm and not be able to solve their problem. In this case, the IVRU server can take one of a number of actions. The IVRU server may connect the user to another network (e.g. the Internet) to search for more information, the user might be put into a voice queue to talk with the next available support engineer to obtain "live" help for their problem, or the user may be allowed to further query to PSS databases.

When a user reaches the lowest level 84 (e.g. as is shown in FIG. 7) and other selective intermediate levels in the hierarchical scheme, the IVRU server may download, register, and execute additional diagnostic applications. Each level in the hierarchical scheme represents a particular type of problem a user may commonly encounter. The additional diagnostic applications are downloaded and launched to collect additional diagnostic information from the user computer and return it to the PSS computer. The additional diagnostic information collected may be used to further direct graphical information to the user, or be sent to a support engineer who will provide "live" help.

As one specific example, using the word processing printer problem described above, when the user arrives at the lowest level in the hierarchical scheme (FIG. 7), a diagnostic interpreter (e.g. a Printer Control Diagnostic Interpreter) on the PSS computer would be launched. The Printer Control Diagnostic Interpreter would then download (i.e. if the Printer Control Diagnostic Agent did not already exist on the user computer) and execute a diagnostic agent (e.g. a Printer Control Diagnostic Agent) on the user's computer. The Printer Control Diagnostic Agent would then run diagnostics on the user's printer, check the printer related software on the user computer, etc. using other pairs of specific diagnostic agents and diagnostic interpreters.

The diagnostic information collected by the Printer Control Diagnostic Agent on the user computer is then sent back to the Printer Control Diagnostic Interpreter on the PSS computer. The Printer Control Diagnostic Interpreter saves the information for later display and interpretation by a support engineer. A similar sequence takes place for network problems, application problems, etc.

In many cases, the user may solve their own problem by navigation through the graphical representation of the interactive voice response menu. As a result, the user may terminate the connection before talking to a support engineer. This may save the creator of the computer software significant support costs. However, if the user cannot solve their problem based on navigation through the graphical interactive voice response menu, or using queries to PSS databases, the user is presented with a message that instructs them to stay on the line and they will be able to talk with a support engineer. The user is then added to a voice queue.

When a support engineer becomes available, data collected by any additional diagnostic agent(s) launched during the user's navigation through the hierarchical scheme, if any, is displayed for the support engineer (as was described above for the printer problem example).

The support engineer can quickly examine the data and discuss the results with the user. If additional diagnostic information is required, the support engineer can download and execute additional diagnostic applications to collect the information from the user computer. As a result, the user's problem may be diagnosed more quickly and thoroughly than it could by using other methods such as an interactive voice response menu or voice interaction with a support engineer.

In another embodiment of the invention, a user can connect to the support center by making a data connection. If a data connection is made (e.g. with a generic modem, Internet connection, etc.), no voice communications are available. This allows a wide variety of users without a VoiceView™ modem to also interact with the support center.

In this embodiment, the voice interactive response menu decision tree is automatically displayed in a graphical format as the default option. The user navigates through the graphical interactive voice response menu as described above. This may include having the PSS computer launch diagnostic applications on the user computer when the user visited selected levels in the hierarchical scheme. However, launching diagnostic applications on the user computer may depend on what type of data connection has been established between the user computer and the PSS computer.

Since a user does not have a voice connection to

talk to a support engineer for further assistance, if the user requires further assistance, the user would have to terminate the data connection, and establish a voice connection or voice/data connection with the support center. However, even without a voice connection, the user can still receive the diagnostic benefits associated with navigation through the graphical interactive voice response menu described above.

Having illustrated and described the principles of the present invention in a preferred embodiment, it should be apparent to those skilled in the art that the embodiment can be modified in arrangement and detail without departing from such principles. Accordingly, I claim as my invention all such embodiments as come within the scope and spirit of the following claims and equivalents thereto.

The features disclosed in the foregoing description, in the claims and/or in the accompanying drawings may, both, separately and in any combination thereof, be material for realising the invention in diverse forms thereof.

Claims

1. In a system having a first computer coupled to a remote second computer by a communications circuit, each computer having a central processing unit coupled to a memory, a display for visually outputting data signals, a speaker for outputting audio signals, the remote second computer having one or more interactive voice response units, the communication circuit allowing the transfer of both audio signals and data signals, a method of obtaining responses to a set of queries comprising the steps of:

   sending a first set of queries from the first computer to the second computer, the first set of queries comprising an audio signal allowing a user of the second computer to select display of subsequent queries on the second computer display;
   outputting the first set of queries on the second computer using the speaker;
   monitoring the communications circuit from the first computer to detect a specific response to the first set of queries from the second computer; and
   in response to the detection of said specific response,
   sending a subsequent set of queries comprising data signals to the second computer, and
   outputting the subsequent set of queries using the display on the second computer.

2. The method of claim 1 where sending a first set of queries includes sending the first set of queries using an interactive voice response unit.

3. The method of claim 1 where the monitoring step includes monitoring the communications circuit for both audio and data signals.

4. The method of claim 1 wherein the specific response is a telecommunications signal including DTMF tones.

5. The method of claim 1 where the outputting the subsequent set of queries step includes outputting query data using a hierarchal paradigm.

6. The method of claim 5 where the hierarchal paradigm is a decision tree.

7. In a system having a first computer coupled to a remote second computer by a communications circuit, each computer having a central processing unit coupled to a memory, a display for visually outputting data signals, a speaker for outputting audio signals, the remote second computer having one or more interactive voice response units, the communications circuit allowing the transfer of both audio signals and data signals over said communications circuit, a method of displaying a plurality of queries:

   maintaining a database on the first computer containing database entries listing a plurality of interactive voice response queries;
   consulting the database on the first computer to obtain a current plurality of interactive voice response queries to display on the second computer;
   sending a first subset of said current plurality of interactive voice response queries to the second computer;
   outputting said subset of interactive voice response queries on the second computer;
   allowing a user to selectively respond to any of the outputted subset of interactive voice response queries;
   monitoring the communications circuit to detect a response to the subset of interactive voice queries from the second computer; and
   upon detecting a response, sending a subsequent subset of the plurality of interactive voice response queries to the second computer for display based on the user responses to said first subset of queries.

8. The method of claim 7 where the consulting step includes using data obtained from previous interactions by users who have had computer software problems.

9. The method of claim 7 where the sending steps include sending the subset of interactive voice response queries as data signals.

10. The method of claim 7 where the sending steps include sending the subset of interactive voice queries as audio and data signals.

11. The method of claim 7 where the outputting the subset of queries step includes outputting said interactive voice response queries using a hierarchal paradigm.

12. The method of claim 11 where the hierarchical paradigm is a decision tree.

13. The method of claim 7 where the outputting step includes outputting the interactive voice queries on the display.

14. The method of claim 7 where the allowing step includes allowing a user to jump to any subset of interactive voice response queries which are displayed.

15. The method of claim 7 where the monitoring step includes monitoring the communications circuit for both audio and data signals.

16. In a system having a first computer coupled to a remote second computer by a communications circuit, each computer having a central processing unit coupled to a memory, a display for visually outputting data signals, a speaker for outputting audio signals, the remote second computer having one or more interactive voice response units, the communications circuit allowing the transfer of both audio signals and data signals over said communications circuit, a method of completing automatic remote diagnosis of computer problems comprising the steps of:

maintaining a database on the first computer containing database entries listing a plurality of interactive voice response queries;
consulting the database on the first computer to obtain a subset of the plurality of interactive voice response queries to display;
sending a subset of said plurality of interactive voice response queries from the first computer to the second computer for display;
outputting the subset of interactive voice response queries on the second computer using the display;
monitoring the communications circuit to detect response to the subset of interactive voice response queries from the second computer;
launching a selected diagnostic application on the second computer, the diagnostic application selected from a plurality of diagnostic applications based on the response obtained to the subset of interactive voice response queries from the second computer;
collecting diagnostic information from the second computer using said diagnostic application;
sending said collected diagnostic information from the second computer the first computer; and
analyzing said collected diagnostic information from the second computer on the first computer.

17. The method of claim 16 where the monitoring step includes monitoring the communications circuit for user inputs from the second computer.

18. The method of claim 16 where the launching step includes downlcading a diagnostic application from the first computer to the second computer and executing said diagnostic application on the second computer.

19. The method of claim 16 where the launching step includes executing resident a diagnostic application on the second computer.

20. In a system having a first computer coupled to a remote second computer by a communications circuit, each computer having a central processing unit coupled to a memory, a display for visually outputting data signals, the remote second computer having one or more interactive voice response units, the communications circuit allowing the transfer data signals over said communications circuit, a method of graphically displaying a plurality of interactive voice response queries comprising:

maintaining a database on the first computer containing database entries listing a plurality of interactive voice response queries;
consulting the database on the first computer to obtain a first subset of the plurality of queries to graphically display on the second computer;
sending the first subset of said current plurality of queries to the second computer;
outputting graphically on the second computer said first subset of queries;
allowing a user to selectively respond to any of the outputted first subset of queries;
monitoring the communications circuit on the first computer to detect a response from the from the second computer; and
upon detecting a response, selecting a second subset of queries for sending to the second computer for graphical display, the second subset of questicns being selected based on the user responses to said first subset of queries.

FIG. 1

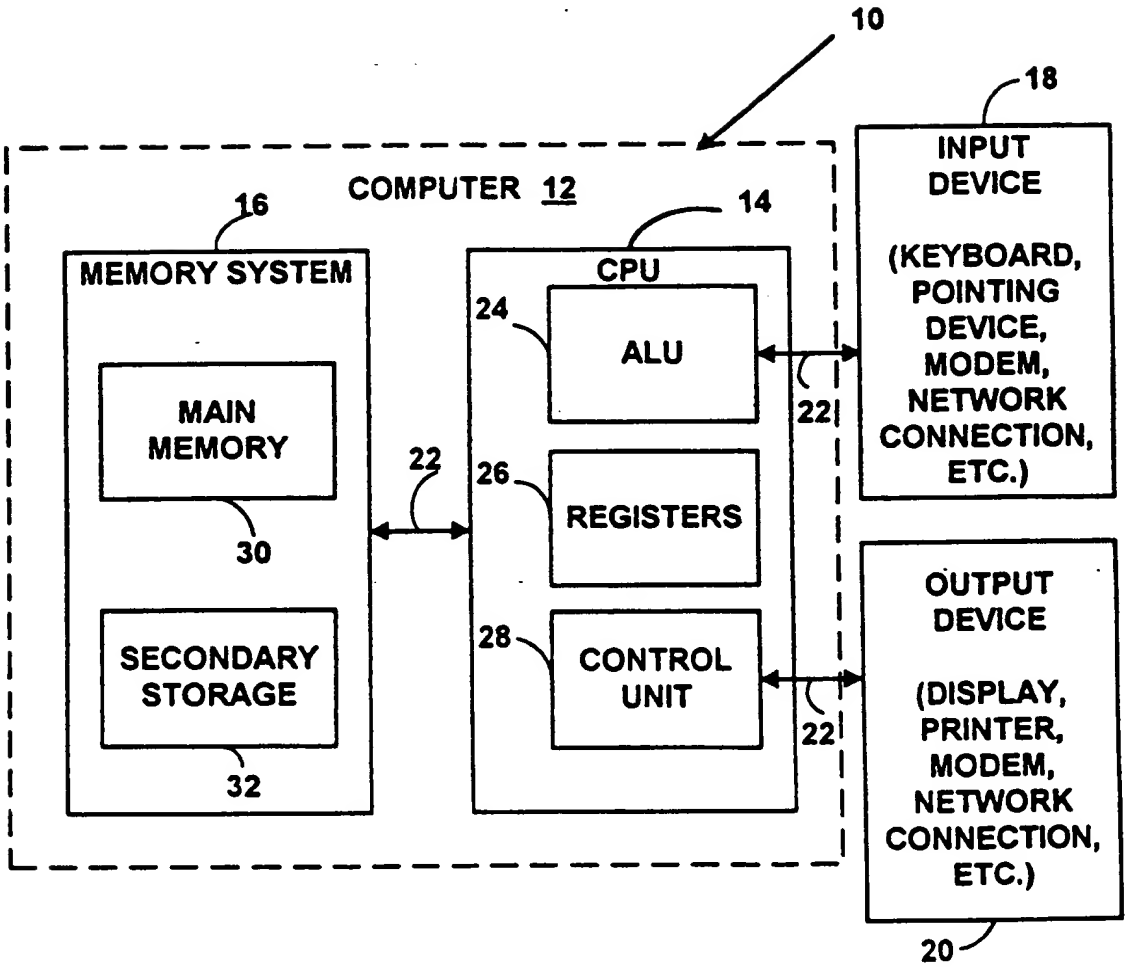

10
18
INPUT DEVICE
(KEYBOARD, POINTING DEVICE, MODEM, NETWORK CONNECTION, ETC.)
COMPUTER 12
16
MEMORY SYSTEM
MAIN MEMORY
30
SECONDARY STORAGE
32
22
14
CPU
24
ALU
26
REGISTERS
28
CONTROL UNIT
22
22
OUTPUT DEVICE
(DISPLAY, PRINTER, MODEM, NETWORK CONNECTION, ETC.)
20

FIG. 2

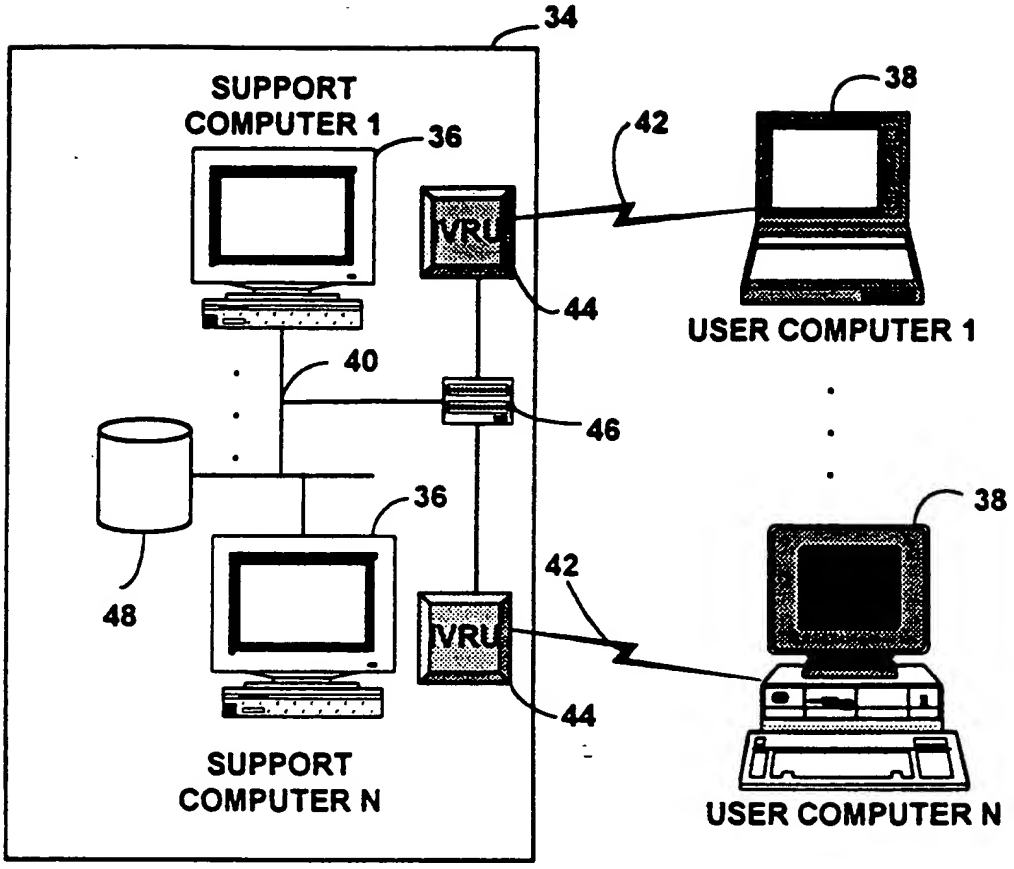

34
SUPPORT
COMPUTER 1
36
VRU
42
38
44
USER COMPUTER 1
40
46
36
38
48
42
VRU
44
SUPPORT
COMPUTER N
USER COMPUTER N

FIG. 3

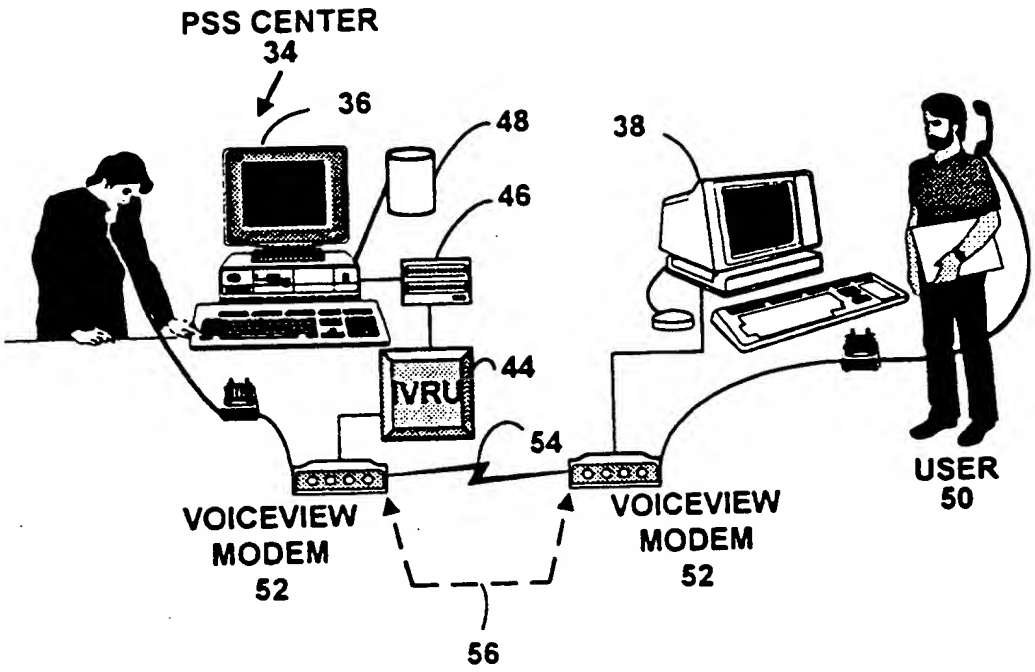

PSS CENTER
34
36
48
46
38
44
VRU
54
VOICEVIEW
MODEM
52
VOICEVIEW
MODEM
52
USER
50
56

FIG. 4

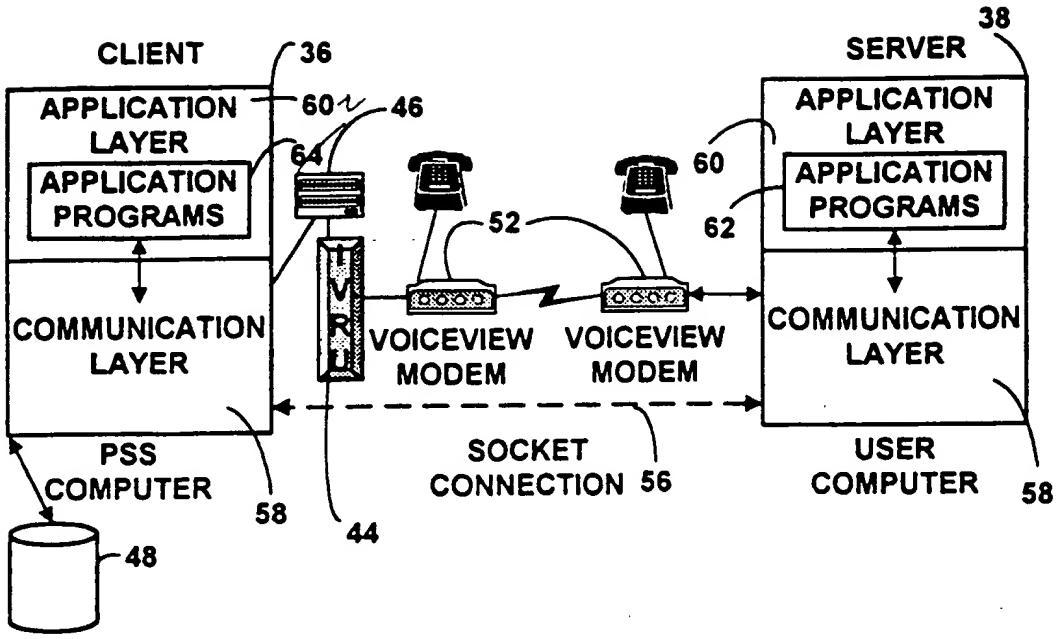

CLIENT
36
SERVER
38
APPLICATION LAYER
60
46
APPLICATION PROGRAMS
64
52
60
APPLICATION LAYER
APPLICATION PROGRAMS
62
COMMUNICATION LAYER
IVRU
VOICEVIEW MODEM
VOICEVIEW MODEM
COMMUNICATION LAYER
PSS COMPUTER
58
SOCKET CONNECTION
56
USER COMPUTER
58
44
48

FIG. 5

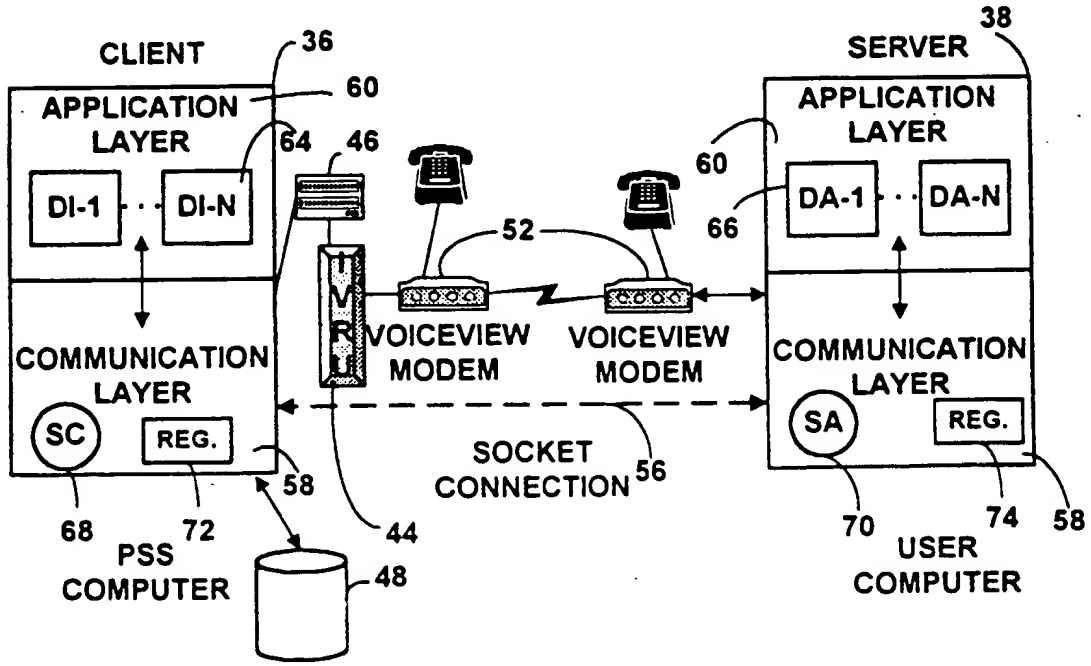

CLIENT
36
SERVER
38
APPLICATION LAYER
60
APPLICATION LAYER
64
46
60
DI-1
DI-N
DA-1
DA-N
66
52
IVRU
VOICEVIEW MODEM
VOICEVIEW MODEM
COMMUNICATION LAYER
COMMUNICATION LAYER
SC
REG.
SA
REG.
58
SOCKET CONNECTION
56
68
72
70
74
58
PSS COMPUTER
44
48
USER COMPUTER

FIG. 6A

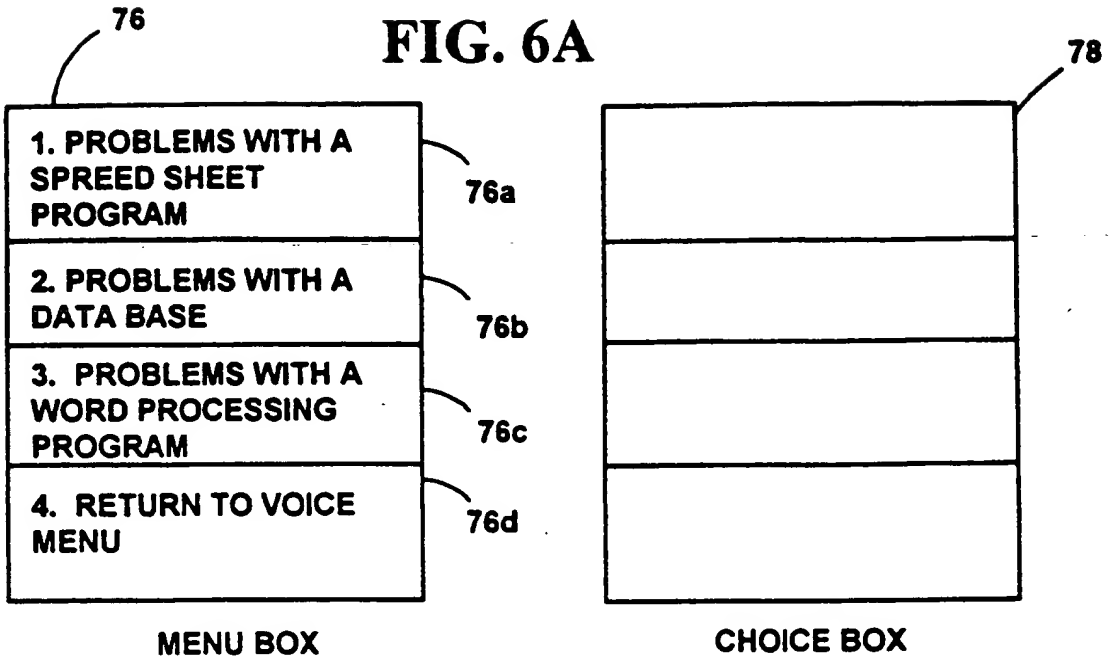

76
1. PROBLEMS WITH A SPREED SHEET PROGRAM
76a
2. PROBLEMS WITH A DATA BASE
76b
3. PROBLEMS WITH A WORD PROCESSING PROGRAM
76c
4. RETURN TO VOICE MENU
76d
78
MENU BOX
CHOICE BOX

FIG. 6B

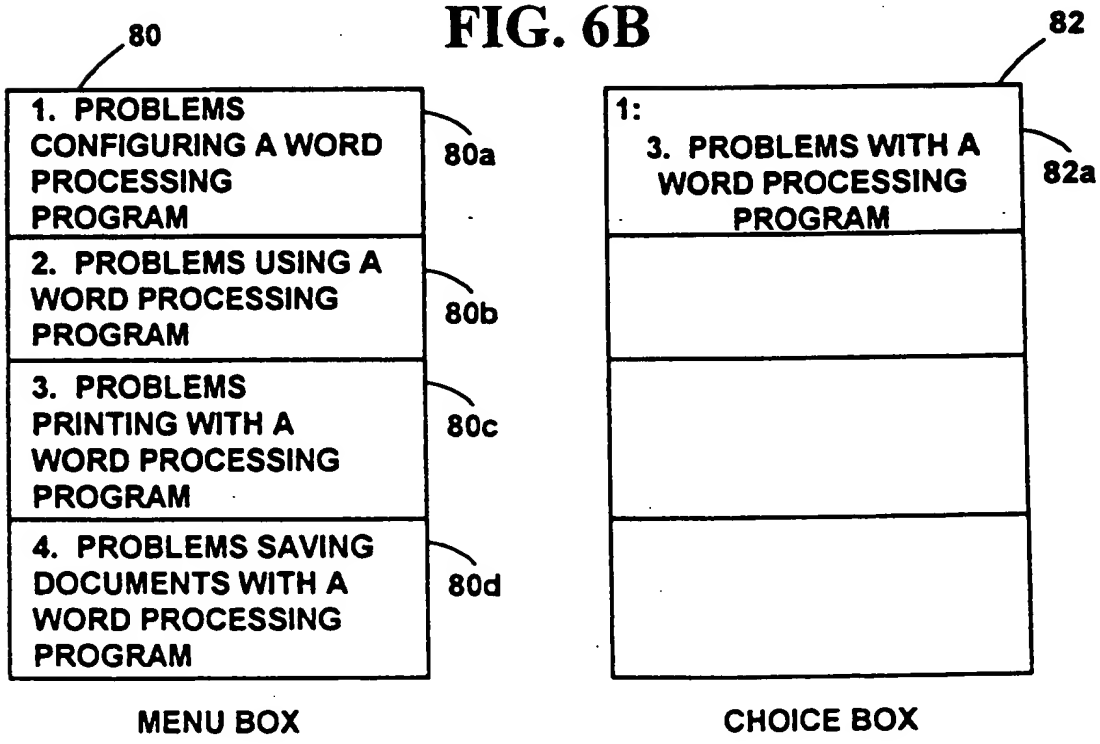

80
1. PROBLEMS CONFIGURING A WORD PROCESSING PROGRAM
80a
2. PROBLEMS USING A WORD PROCESSING PROGRAM
80b
3. PROBLEMS PRINTING WITH A WORD PROCESSING PROGRAM
80c
4. PROBLEMS SAVING DOCUMENTS WITH A WORD PROCESSING PROGRAM
80d
82
1:
3. PROBLEMS WITH A WORD PROCESSING PROGRAM
82a
MENU BOX
CHOICE BOX

FIG. 7

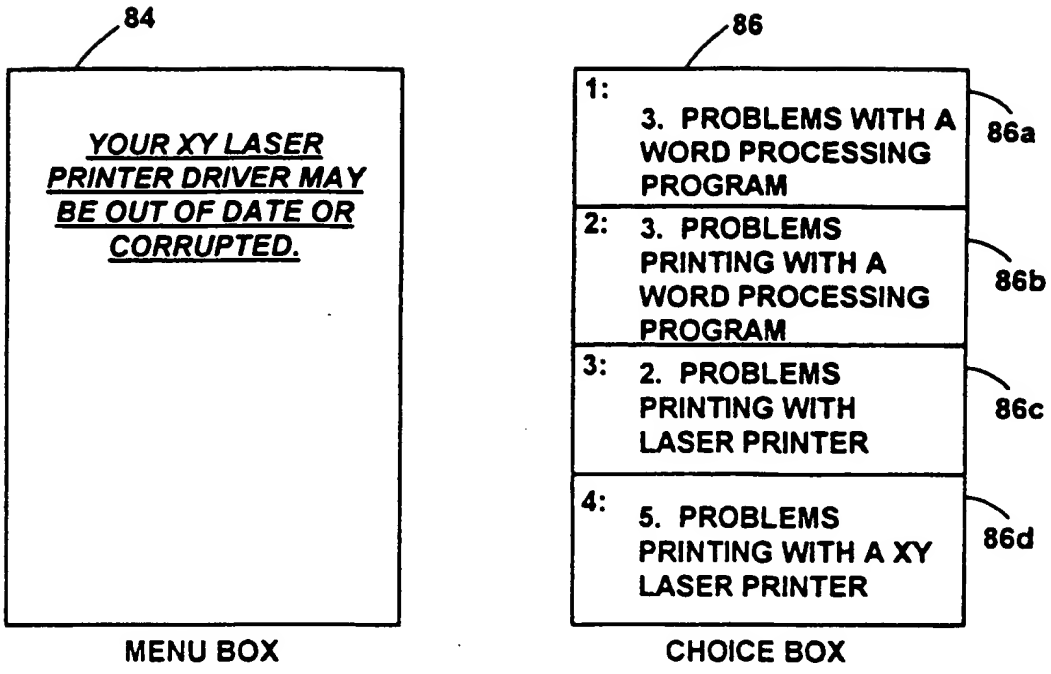
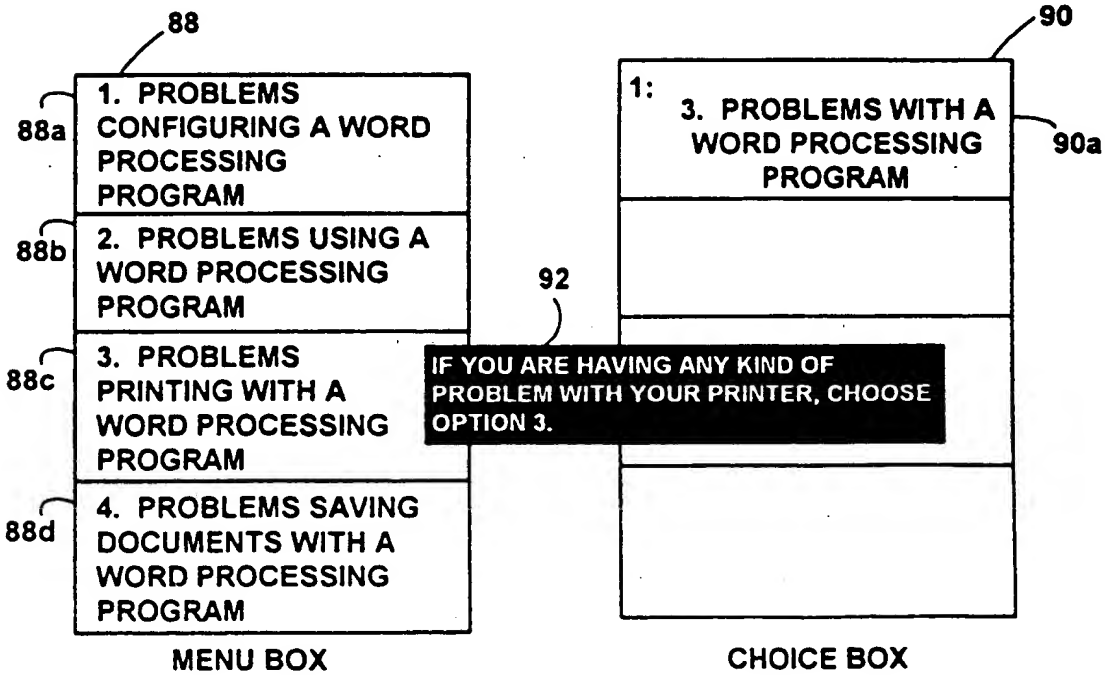

84
YOUR XY LASER PRINTER DRIVER MAY BE OUT OF DATE OR CORRUPTED.
MENU BOX
86
1: 3. PROBLEMS WITH A WORD PROCESSING PROGRAM
86a
2: 3. PROBLEMS PRINTING WITH A WORD PROCESSING PROGRAM
86b
3: 2. PROBLEMS PRINTING WITH LASER PRINTER
86c
4: 5. PROBLEMS PRINTING WITH A XY LASER PRINTER
86d
CHOICE BOX

FIG. 8


88
88a
1. PROBLEMS CONFIGURING A WORD PROCESSING PROGRAM
88b
2. PROBLEMS USING A WORD PROCESSING PROGRAM
88c
3. PROBLEMS PRINTING WITH A WORD PROCESSING PROGRAM
88d
4. PROBLEMS SAVING DOCUMENTS WITH A WORD PROCESSING PROGRAM
MENU BOX
92
IF YOU ARE HAVING ANY KIND OF PROBLEM WITH YOUR PRINTER, CHOOSE OPTION 3.
90
1: 3. PROBLEMS WITH A WORD PROCESSING PROGRAM
90a
CHOICE BOX